Computer Systems Science & Engineering
DOI:10.32604/csse.2021.014541
Article

Tech Science Press

Data Security Storage Model of the Internet of Things Based on Blockchain

Pingshui Wang[1,2,*] and Willy Susilo[2]

[1]School of Management Science and Engineering, Anhui University of Finance and Economics, Bengbu, 233030, China
[2]School of Computing and Information Technology, University of Wollongong, Wollongong, 2522, Australia
[*]Corresponding Author: Pingshui Wang. Email: 120081049@aufe.edu.cn
Received: 27 September 2020; Accepted: 20 October 2020

Abstract: With the development of information technology, the Internet of Things (IoT) has gradually become the third wave of the worldwide information industry revolution after the computer and the Internet. The application of the IoT has brought great convenience to people's production and life. However, the potential information security problems in various IoT applications are gradually exposed and people pay more attention to them. The traditional centralized data storage and management model of the IoT is easy to cause transmission delay, single point of failure, privacy disclosure and other problems, and eventually leads to unpredictable behavior of the system. Blockchain technology can effectively improve the operation and data security status of the IoT. Referring to the storage model of the Fabric blockchain project, this paper designs a data security storage model suitable for the IoT system. The simulation results show that the model is not only effective and extensible, but also can better protect the data security of the Internet of Things.

Keywords: Internet of Things (IoT); blockchain; data security; digital signatures; encryption; model

1 Introduction

The Internet of Things (IoT) refers to a huge intelligent network based on the communication of things formed by combining information devices such as sensor networks, radio frequency reading devices, bar codes, global positioning systems and infrared sensors with the Internet through various access networks. The IoT has realized the connectivity between things and the communication between people and things, which will become an important basic platform for the information society in the future. Its application has developed from military reconnaissance, environmental monitoring to intelligent home, smart city, intelligent medical care and other fields closely related to human life. However, the ubiquitous interconnection of things and data exchange between people and things also put forward higher security requirements for the collection, transmission and storage of sensitive information. Due to the limitation of its own resources, the IoT has an unprecedented opportunity for development. At the same time, there are some security problems that need to be solved urgently.

The security of the IoT can be divided into application security and sense security [1]; the former refers to the traditional data security, service, storage security and network security, etc.; the latter refers to the security of network communication between perception nodes or terminals. Since perception nodes are often used as the source of information, their geographical location is very important in location-based service networks. In recent years, the research on location privacy technology has received extensive attention from the academic community. Many security models and privacy protection methods have been proposed [2–6].

In the IoT, data belonging to one organization can be equally valuable to members of another. If the data are stored and maintained independently from each other and cannot be circulated, an island of data will be formed, which is not conducive to exploiting the value of data. A common solution is that multiple organizations build a data center together. Due to the high cost and maintenance difficulty of self-built data centers, third-party cloud platforms are often used at present. This model gives the data to the centralized cloud platform for management. The data sharing is uniformly scheduled by the cloud platform, which greatly improves the efficiency and avoids data inconsistency. However, this model also has some problems [7].

(1) *Transmission delay and resource waste*. Because the data is stored in the cloud, even if the data within the organization is also requested to access the cloud, which causes unnecessary delay. If the data center is built locally, the problem of duplicate construction will arise.

(2) *Failure of single point*. Data reading and writing depend on the cloud platform. The centralization of the network structure puts great pressure on the cloud platform. Once it fails, the system will be paralyzed.

(3) *Security and privacy protection*. Uploading data to the cloud platform means that it has the ability to access the data as a third party. However, such centralized institutions are not completely credible in many cases. Managers of the cloud platform or external personnel may tamper with or disclose the data, thus damaging the integrity and confidentiality of the data.

In view of this, it is of great practical significance to study the secure and efficient de-centralized data storage model of IoT.

However, due to the complex topology structure of the IoT and the resource constraints of many devices in the network, some traditional security technologies are not fully applicable to the IoT, and many related protection technologies still need to be broken through [8].

Blockchain is a distributed ledger technology which is maintained by multiple parties and whose data cannot be changed. It is characterized by decentralization, joint maintenance, non-tampering, encryption and security. It allows participants to build trust without being part of a centralized institution, which is suitable for improving the existing architecture of the IoT [9].

This paper introduces the blockchain technology and combines it with the IoT to propose a data security storage model, which can effectively solve the data security problem of the IoT. For example, due to the existence of sensitive data generation, exchange and storage between devices in the IoT, its privacy, robustness and single-point fault tolerance can be ensured by virtue of the P2P characteristics of the blockchain. Each operation of data creation, modification and deletion in the IoT can be registered and verified in the blockchain to realize the detection of data tampering and abuse in the IoT.

The rest of this paper is organized as follows. In Section 2, we introduce the related work. In Section 3 we present our data security storage model. In Section 4, we analyze the performance of our method through extensive experiments. Section 5 contains the conclusion and future work.

2 The Related Work

Currently, most IoT solutions rely on a centralized server-client paradigm to access IoT devices in the form of cloud computing. In this centralized model, the linear growth of centralized cloud computing

capacity cannot match the explosive growth of devices and data scale; the transmission bandwidth of network edge equipment and cloud server limits the network performance, and the network delay also brings more security problems. The centralized data management makes the privacy security problem become prominent [10]. Data transmission of network edge devices and cloud servers brought IoT equipment large overhead [11]. Therefore, there appeared some IoT architectures based on distributed P2P [12], which can solve the problems existing in the centralized network architecture. However, until the advent of blockchain technology and combining with the IoT, many problems of the IoT are effectively solved. In the frame of the blockchain, IoT devices can safely store their data in a different node without human intervention, and ensure its decentralized trust, authenticity, security, and privacy using the blockchain features.

Blockchain is a point-to-point distributed ledger technology based on a cryptography algorithm [13]. For the first time, blockchain solves the security problem caused by trust-based centralization model. It ensures the secure transfer of data and value based on the cryptography algorithm, the traceability and unforgeability of data based on the hash chain and timestamp mechanism, and the consistency of block data among nodes based on consensus algorithm.

Currently, blockchain technology has been used in a variety of IoT scenarios, involving sensors, data storage, identity management, timestamp services, wearable devices, supply chain management and other technologies, which also covers agriculture, finance, medical care, transportation and other fields.

Most of the applications of blockchain technology in the IoT in the academic circle focus on how to use blockchain technology to solve the problem of a high degree of centralization of the IoT and how to develop the corresponding applications of the IoT with the blockchain as the supporting platform.

Ouaddah et al. [14] for the first time put forward using blockchain to ensure the security of IoT data privacy—FairAccess, using the consistency of distributed books to solve the centralized and distributed access control problems in the IoT, which opened up a new field of blockchain application, namely the IoT access control. Pinno et al. [15] proposed the ControlChain, which is a kind of access control architecture based on the blockchain. This architecture can not only protect the privacy of IoT, but also be compatible with all kinds of access control model of IoT. Cha et al. [16] proposed a design of blockchain linking gateway, which can be adaptive to protect user privacy. Blockchain plays an intermediary role between the user and the IoT devices. The user can get equipment information and privacy policy connected to the blockchain gateway. The design of the blockchain gateway makes a contribution to improving user privacy security and IoT application trust without changing the traditional structure of IoT. However, even under the blockchain structure, if the data is encrypted through symmetric key encryption such as the advanced encryption standard (AES), in the process of miners demonstrate block, key and the data will be shared together, which means that the data privacy of blockchain IoT is still a problem. Rahulamathavan et al. [17] proposed a privacy blockchain IoT architecture based on the attribute-based encryption (ABE), in which ABE was applied to blockchain to reconstruct the blockchain agreement, which provides an end-to-end privacy protection scheme for the Internet ecosystem, and offers a new way to solve the problem of practical application. However, privacy optimization based on the encryption algorithm also brings more computational overhead for IoT devices. The pros and cons need to be weighed against the practical application scenarios.

Yeow et al. [18] proposed that the combination of blockchain and other technologies can be effectively applied to the edge computing architecture of the IoT to solve the privacy and security problems.

Dorri et al. [19] proposed an optimized blockchain, which can solve the problems of high resource consumption, low expansibility and long processing time in the classic blockchain, while maintaining security and privacy for the IoT.

Ding [20] analyzed the characteristics and security problems of the IoT application, proposed to solve the possible security problems under the current centralized design architecture of the IoT with decentralized blockchain technology, and designed a computing method for the ID of devices in the IoT. It provides constructive suggestions for the decentralized structure of the IoT, which is mainly to improve the security architecture of the IoT, but does not involve the detailed security protection technology based on the blockchain.

Yuan et al. [21] took BitCoin transaction as an example to elaborate on the basic model of blockchain and the characteristics of key technologies at all levels of the blockchain system, which provides useful guidance and reference for future studies related to the IoT. However, the paper did not involve specific applications other than a digital currency, but let alone linking the blockchain technology with the information security of the IoT from the perspective of defending against illegal attacks.

3 Design of Data Security Storage Model Based on Blockchain

3.1 Model Overall Design

Distributed ledger technology as the basic blockchain technology, ensures that the IoT data storage service center can still provide normal services after single point failure by using Raft consensus protocol [22]. However, the Raft consensus protocol with strong consistency can only guarantee the consistency of the data in the distributed storage center, cannot guarantee the integrity and untamperability of the data. Therefore, we need to introduce the blockchain data structure and hash function [23].

Considering that the computing power of nodes in the IoT is uneven, and some physical nodes do not have enough space to store data information, we designed a hierarchical storage structure based on blockchain, as shown in Fig. 1.

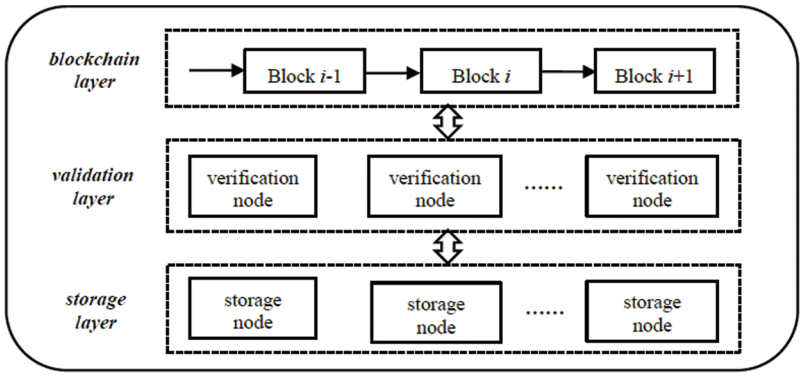

Figure 1: Data storage hierarchy of the IoT based on blockchain

The whole storage structure is divided into three levels: *blockchain layer*, *validation layer* and *storage layer*. The *blockchain layer* is divided into global blockchain and local blockchain. The global blockchain is jointly maintained by the whole IoT network, which represents the latest state of the network. Each node ensures that the local state is always up to date by synchronizing the local blockchain state. The *validation layer* is used to guarantee the security of the storage system. When new message storage requirements appear in the network, the verification node group will verify the storage location information. According to the verification strategy, the data can only be stored and the blockchain can be updated when more than a certain proportion of verification nodes retrieve the same copy of the storage location information. The *storage layer* is responsible for storing specific data. The storage layer nodes

generally have a large storage space, while the other physical nodes only record copies of data storage location information and local blockchain state.

Combining with the hierarchical storage structure, we design the information security storage process of the IoT based on the blockchain, which can be divided into the following steps.

1. The blockchain project Fabric CA is used to distribute key pairs, including a public key and private key, for entity nodes of the IoT.
2. The message content and public key of the sender node i are taken as input to generate the message verification code using Eq. (1).

$$\mathbf{HM}(\boldsymbol{PK_i}, \boldsymbol{M_i}) = \mathbf{H}(\boldsymbol{PK_i}|\boldsymbol{M_i}) \tag{1}$$

In which, $\boldsymbol{PK_i}$ is the public key of the sender node i, $\boldsymbol{M_i}$ is the message content and H is the hash function.

1. Use the private key of the sender node to generate the signature of the message verification code, store the signed message to the storage node, and generate a copy of the storage location information, including ledger number and block number.
2. The verification layer verifies the storage location of the data according to the default verification strategy of Raft consensus protocol. If the verification is successful, it goes to Step (5). Otherwise, the data is discarded and the procedure of information storing is terminated.
3. Generate metadata based on the signed message and the copy of storage location information.
4. Hash the generated metadata to get the block body.
5. The block header is generated based on the hash value, timestamp and block height information of the previous block, which is combined with the block body to form the block.
6. Record the new block to the global blockchain, and inform all the nodes to synchronize the local blockchain state, so as to complete the storage of IoT information.

Referring to the blockchain project, a storage layer security storage structure is designed to facilitate the data operation of the IoT, as shown in Fig. 2.

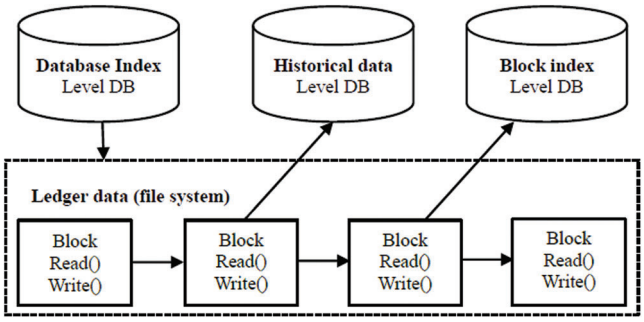

Figure 2: Distributed ledgers of IoT data

The file system that actually stores the data information of the IoT is called a ledger. Each physical node of the storage layer can maintain multiple ledger books according to actual needs, classify the data in the IoT, and store the data into different ledger books in the form of blockchain. Each ledger book contains the following necessary information.

1. The ledger number: Store data according to its type, and quickly query the ledger book.
2. Ledger data: Block data stored in form of the file.

3. Status data: The latest global status information, which is used to synchronize data between nodes.
4. Historical data: The historical version of the status data, which is used to restore the data.
5. Block index: Fast query block, that is, the IoT data contained in the block.

The ledger number is used to distinguish which ledgers exist in the IoT. It does not keep the data related to the block. When creating a new ledger, it is necessary to check whether the same ledger number exists to ensure the uniqueness of the global ledger.

The ledger data is stored in the node file system in binary form, and each ledger data is stored in a different directory depending on the number of the ledger.

The Level DB database of key-value type is used to store the state data. Each physical node will maintain four DBs, namely: *ID_Store* DB for storing the blockchain ID; *State* DB for storing global state; *Block* DB for storing blocks; *Version* DB for storing changed key.

3.2 Design of Data Block Structure of the IoT

In this paper, the blockchain structure is introduced to provide the integrity and efficiency of the secure storage of IoT data by using a hash function, Merkel tree, timestamp and other technologies, which ensures that the stored data cannot be tampered with. Based on the Fabric block structure [24], the block structure suitable for the IoT system is designed, as shown in Fig. 3.

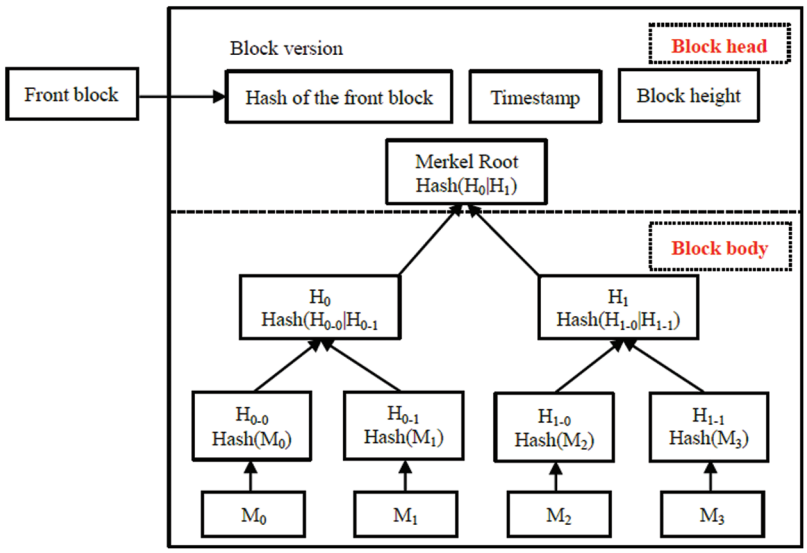

Figure 3: Data block of IoT

The block of IoT is divided into *block head* and *block body*. The *block body* is a Merkel-tree, and the non-leaf node is a message digest generated by a hash function. In this paper, a hash function SHA256 is used to hash the data content. Hash algorithms can map arbitrary length binary plaintext strings to shorter binary strings (hash digests), and for good hash functions, different plaintext is difficult to calculate the same digest, called collision resistance. Any change of block body M_0–M_3 will lead to a difference in the hash digest calculated by SHA256, and thus the Root content will change, so that it cannot be linked by the block. In addition, as can be seen from Fig. 3, each block has the hash value of the front block, so each block provides a hash value to verify whether the data contained in this block was changed, which can effectively prevent the block body data from being tampered with. At the same

time, the introduction of the digest enables users to verify the validity of the data without knowing the specific content of the data itself.

The *block header* contains the block version number, the hash value of the front blocks, timestamp, and the block height. For a blockchain with a height of K, if an attacker wants to change the data of block unit i, due to its successive blocks $i + 1$ holds the hash digest of block unit i, based on the collision resistance of hash function, it is easy to find the inconsistency with the $i + 1$ saved hash digest through calculating the hash digest of i, and judge the data was malicious tampering. Once the tampered position is positioned, the evil behavior of the attacker is easy to find. And because there is a global blockchain, tampering does not work.

Data blocks of the IoT are designed according to the Merkel tree, in which the root is calculated by Hash (left|right) on the left and right subtrees of the Merkel tree using a hash algorithm SHA256. Meanwhile, the hash values of all nodes of the Merkel tree are calculated by hash algorithm SHA256.

3.3 Block Files Manager of the IoT

The functions of the block files manager can be divided into two parts: Ledger data storage management and block index management.

Ledger data storage management mainly realizes the access of IoT data and determines the storage directory of block files and the location of block storage files. Block index management is mainly used to provide fast block data query services.

3.3.1 Ledger Data Storage Management

(1) Ledger Data Storage

At present, the default file size of the ledger block of Fabric is 64 MB, and the maximum amount that a ledger can hold is about 61 TB, which meets the requirements of practical application scenarios of the IoT. According to the naming rules of the Fabric block file, set the filename of the block to be created with "IOTblockfile_" as prefix and 6-digit as the suffix. The suffix number of the file must be continuous without any missing, such as IOTblockfile_000001, IOTblockfile_000002, IOTblockfile_000003, etc.

The IoT block manager maintains a pointer that writes to a block file, which includes both the block size and the block data. When the size of the block file exceeds the set value (maximum 64 MB), the data will be written to the next block file.

If any exception occurs during the writing of the block, the block file will be restored to its pre-write state. Because each block file contains multiple blocks, the block index information needs to be recorded outside the block file so that it can be quickly located and searched for blocks. The block index will be stored in the database and eventually, the blockchain information will be updated globally.

(2) Ledger Data Reading

For the reading of block data, Fabric blockchain adopts the way of data flow, which is divided into the IoT block file flow and block flow. Block file flow can read block information sequentially from the given offset address to the end of the file in a single file, and define block flow on the basis of block file flow. Block flow can read block information sequentially from multiple files, starting from the offset of a given file to the end of the block number of the file.

3.3.2 Block Index and Status Database

In order to find the block easily, quickly verify the correctness of the block data and synchronize the block data, two block index methods are provided: An index based on the block number and an index based on the block hash. The index is a file location pointer consisting of the number of the file in which it is located, the offset within the file, and the block size in bytes.

According to the block index design, block lookup is a three-level index process, as shown in Fig. 4. First, determine the ledger ID to look for, then find the corresponding block file according to the block file number, and finally, extract the block content according to the offset and the size of the block bytes.

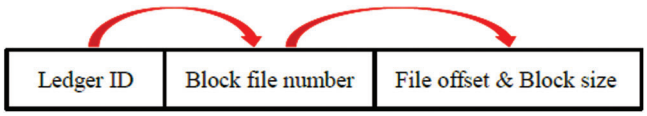

Figure 4: Three-level index of blocks under multiple ledgers

The status data records the latest global status, including the latest information transmission execution results in the IoT, which is stored in the status database. The state database is implemented by Level DB, and the basic operation is based on key-value pairs. As the log index view of information transmission transaction in the IoT, the status data can be generated according to the blockchain at any time, and automatically restored by the global blockchain when the node in the IoT starts. Different ledgers in the IoT correspond to different state databases and are stored in different directories.

In the Fabric blockchain network, the state database records two checkpoint information: index checkpoint and block checkpoint. The index checkpoint records the number of the last indexed block and represents the latest state of records in the database, while the block checkpoint information records a state of the block file system. The two checkpoint information may not be up to date and merely represent the correct state of the ledger at some point in the past, so there is a checkpoint information synchronization process before the stored procedure for the ledger data to ensure that the global state data is up to date.

According to the previous introduction of the block stored procedure, the block recording process is to submit the block to the block file system first, then record the block checkpoint information, and finally record the index checkpoint information. Therefore, checkpoint information can be restored and updated through the block file system. The recovery and update process of two checkpoints information is called the checkpoint synchronization process.

The block file flow mentioned above will be used in the synchronization process, which determines whether there is a block after the checkpoint information, and finds the largest block number to complete the synchronization process.

3.4 Historical Data

The historical data is used to record every message transmission within the network, but only the history of message transmission is recorded, not the message content. It is the historical version of the status data and stored in the Level DB database. Only the key of a valid transaction is stored, instead of the corresponding value. Each key is a quad: Namespace, written key value, block number, and transaction number.

1. Namespace: Represents the different chain in the network, the data of different blockchain is logically isolated.
2. Written key value: The key value of data to be written.
3. Block number: The block number of the data to be written.
4. Transaction number: The number assigned by PBFT in the data transmission process of the IoT is called the transaction number to be written into the block.

By querying the data content of the historical database, information transmission records between any nodes in the network can be known, as well as information such as block height and transaction number when storing data.

In addition to recording the network's own data storage behavior, the historical database will also record the user's operation records on the ledger. Considering the characteristics of blockchain structure and security, the ledger is only provided data query service to the users. Therefore, the network administrator can check the historical data of ledger operation to find out whether there is unreasonable query data behavior of users and whether there is an illegal query of users, and make a timely response to prevent more severe economic losses caused by data leakage.

4 Experimental Analyses

4.1 Security Verification

The experiment will be divided into two parts to test and analyze the performance of the block-based secure storage technology designed in this paper. Firstly, the data security under the common storage technology is tested for an attack, then the storage structure under the protection of the block structure-based security storage technology designed in this paper is tested for an attack, and finally, the test results are compared and analyzed. The testing environment is a virtual intelligent agricultural data monitoring system, which stores the intelligent agricultural data collected by various sensors.

(1) Use the SQL-Server database to store the data collected by the data collector, and use password authentication and data encryption to protect data security. Then, malicious nodes are introduced and relevant penetration testing tools in the Kali Linux system are used to break the password of the stored database by force.

Experiments show that the password dictionary is used for brute force cracking, and over time, the login username and password are obtained successfully.

The malicious node has obtained the view permission of the database. If the obtained database password has higher permissions, the database structure and data will be exposed completely. If the data is not encrypted, it will be very easy for malicious nodes to achieve the purpose of data theft and data tampering.

In the case of password authentication and encryption of stored data, and the malicious node also gets the username and password after breaking the password. If the attack logs in and find that the database data is encrypted, and thanks to a very secure encryption algorithm, it is almost impossible to decrypt data, then the attack will consider free to tamper with data or even clear the database. Since clearing the database is easy to be discovered by managers, this experiment adopts the more common way of tampering with the data.

By using the obtained password and remotely executing the corresponding database command, the attack can obtain the corresponding table structure in the database.

According to the structure of the table, the update command can be used to change the encrypted data. Since not knowing the specific meaning of the value in the table, the changed data may be invalid, but the attack can achieve the purpose of tampering with the data.

(2) The data collected by the data collector is stored by the secure storage technology based on block structure. Malicious nodes are also introduced to try to attack by using a variety of attack means.

Under the security storage technology designed in this paper, data is stored as a file, and block files are stored in the node file system.

Similarly, the login password of the storage node is obtained by a breaking method. Since the block storage adopts key encryption and hash technology, the data cannot be interpreted. Therefore, the attacker may consider tampering with files or deleting files in order to destroy the normal operation of the system.

First, try to tamper with the data. Because the hash value of the block file storing the data is saved in the next block, if the data is tampered with, the hash value of the whole block will be changed, then the blockchain authentication fails and the data tampering is unsuccessful. Even if the malicious node

tampers with the data of a storage node by advanced attack means, due to the advantages of the structure of the Merkel tree, the location of the malicious modification can be located quickly, and meanwhile, warning information will be sent out by comparing with the data of the storage node.

Then, the more extreme destruction method is used to directly empty the data stored in the storage node. In order to achieve this purpose, the attacker may directly enter the host file system, delete the block file, and make the block file missing.

It can be seen that due to the distributed storage and the existence of global blockchain information, even if the block storage file is deleted or incomplete, the normal work of the whole IoT network will not be affected. The data is restored during the index synchronization process when the node restarts. If the node fails to stop, index synchronization will also be carried out after startup to ensure the node has the latest blockchain state.

However, the host computer enters the data recovery process first after it restarts. During this period, as the local blockchain state is inconsistent with the global blockchain state, storage services will not be provided until the data recovery process is completed and the local blockchain state is synchronized.

Therefore, compared with the traditional storage technology, the distributed storage technology based on block structure not only has better security because of the introduction of blockchain structure, but also the malicious attackers must control multiple storage devices at the same time in order to achieve their goals, which greatly increases the cost and difficulty of their attacks. At the same time, the implementation of distributed storage structure makes the data storage of IoT not affect the normal storage work due to the downtime of a specific machine and improves the robustness of the storage system.

4.2 Storage Efficiency Test

The previous section verified the security of the storage model designed in this paper. The performance of the data security storage technology based on the block structure of the IoT was tested and analyzed in this section.

Considering the problem of node storage space, the storage layer node is composed of a portable computer, PC1, PC2 and PC3, and all nodes participate in verification. Each time the collector sends the collected data, it will be treated as a transaction. The transmitted data will be recorded and constructed into blocks for storage.

Experiments show that verifying the block effectiveness needs about 2.1 ms, storing blocks in the file and updating the status database and global blockchain requires less than 2 s. Meanwhile, after long-term calculation, it is found that the total time spent on generating and saving block files and synchronizing global blockchain will not increase with the increase of block height.

With the increase of data in the network, the number of blocks is also increasing, and the length of blockchain is also getting longer and longer. However, the time spent on increasing blocks does not fluctuate greatly, varies between 200 ms and 235 ms and remains stable about 220 ms, which reflects the advantages of high storage efficiency of block structure.

5 Conclusion and Future Work

In this paper, a secure data storage model for IoT is designed and realized by using a hash function, Merkel tree and blockchain data structure. It can not only ensure the integrity and non-tampering of the stored data, but also avoid the storage of malicious data to some extent because of the existence of the verification layer of the designed storage architecture.

In addition, blockchain structure can only be added and cannot be reduced, so that any data operation in the IoT network will be completely recorded, which provides the data traceability and ensures that any malicious operation of attackers will be exposed, and improves the security of data storage. Digital signature and encryption technology are used in data storage to ensure the privacy of data. Meanwhile, with the increase of the amount of IoT data, the block structure using Merkel tree and hash technology can guarantee the high efficiency of storage, so it can greatly improve the data storage security of the IoT.

However, the security model proposed in this paper does not involve the security of the IoT perception layer and does not constitute a complete IoT security protection system. Therefore, further research can be conducted on whether blockchain-related technologies can provide support for the IoT perception layer and other layers.

Acknowledgement: We thank the anonymous reviewers and editors for their very constructive comments.

Funding Statement: This work was supported by the National Social Science Foundation Project of China under Grant 16BTQ085.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] H. Shen, M. Zhang, H. Wang and W. Susilo, "A lightweight privacy-preserving fair meeting location determination scheme," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3083–3093, 2020.

[2] P. S. Wang, Z. C. Wang, T. Chen and Q. J. Ma, "Personalized privacy protecting model in mobile social network," *Computers, Materials & Continua*, vol. 59, no. 2, pp. 533–546, 2019.

[3] M. Yu, J. Zhang and J. Wang, "Internet of Things security and privacy-preserving method through nodes differentiation, concrete cluster centers, multi-signature, and blockchain," *International Journal of Distributed Sensor Networks*, vol. 14, no. 12, pp. 1–15, 2018.

[4] C. Yin, J. Xi and R. Sun, "Location privacy protection based on differential privacy strategy for big data in industrial Internet of things," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3628–3636, 2017.

[5] S. Wang, Q. Hu and Y. Sun, "Privacy preservation in location-based services," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 134–140, 2018.

[6] W. She, J. S. Chen and Z. H. Gu, "Location information protection model for IoT nodes based on blockchain," *Journal of Applied Sciences-Electronics and Information Engineering*, vol. 38, no. 1, pp. 139–151, 2020.

[7] J. L. Zhang, Y. C. Zhao and B. Chen, "Survey on data security and privacy-preserving for the research of edge computing," *Journal on Communications*, vol. 39, no. 3, pp. 1–21, 2018.

[8] D. Puthal and S. P. Mohanty, "Proof of authentication: IoT-friendly blockchains," *IEEE Potentials*, vol. 38, no. 1, pp. 26–29, 2019.

[9] Z. Y. He, T. Duan and Y. Zhang, "Blockchain in Internet of things: Application and challenges," *Journal of Applied Sciences-Electronics and Information Engineering*, vol. 38, no. 1, pp. 22–33, 2020.

[10] P. K. Sharma, M. Chen and J. H. Park, "A software defined fog node based distributed blockchain cloud architecture for IoT," *IEEE Access*, vol. 6, pp. 115–124, 2018.

[11] Y. J. Ren, Y. Leng and F. J. Zhu, "Data storage mechanism based on blockchain with privacy protection in wireless body area network," *Sensors*, vol. 19, no. 10, pp. 2395–2406, 2019.

[12] D. Y. Kim, A. Lee and S. Kim, "P2P computing for trusted networking of personalized IoT services," *Peer-to-Peer Networking and Applications*, vol. 13, no. 2, pp. 1–9, 2019.

[13] O. Alphand, M. Amoretti and T. Claeys, "IoTChain: A blockchain security architecture for the Internet of things," in *Proc. IEEE Wireless Communications and Networking Conf.*, New Jersey, NJ, USA, pp. 1–6, 2018.

[14] A. Ouaddah, A. A. Elkalam and A. A. Ouahman, "Towards a novel privacy-preserving access control model based on blockchain technology in IoT," in *Europe and MENA Cooperation Advances in Information and Communication Technologies*, Berlin, Germany, pp. 523–533, 2017.

[15] O. J. A. Pinno, A. R. A. Gregio and L. C. E. De Bona, "ControlChain: Blockchain as a central enabler for access control authorizations in the IoT," in *Proc. IEEE Global Communications Conf.*, New Jersey, NJ, USA, pp. 1–6, 2017.

[16] S. Cha, J. Chen and C. Su, "A blockchain connected gateway for BLE-based devices in the Internet of Things," *IEEE Access*, vol. 6, pp. 24639–24649, 2018.

[17] Y. Rahulamathavan, R. C. W. Phan and M. Rajarajan, "Privacy-preserving blockchain based IoT ecosystem using attribute-based encryption," in *Proc. IEEE Int. Conf. on Advanced Networks and Telecommunications System*, New Jersey, NJ, USA, pp. 1–6, 2017.

[18] K. Yeow, A. Gani and R. W. Ahmad, "Decentralized consensus for edge-centric Internet of Things: A review, taxonomy, and research issues," *IEEE Access*, vol. 6, pp. 1513–1524, 2018.

[19] A. Dorri, S. S. Kanhere and R. Jurdak, "Towards an optimized blockchain for IoT," in *Proc. the Second Int. Conf. on Internet-of-Things Design and Implementation*, New Jersey, NJ, USA, pp. 173–178, 2017.

[20] P. Ding, "Research and application of Internet of Things security based on decentralization," M.S. Theses, Beijing University of Posts and Telecommunications, China, 2018.

[21] Y. Yuan and F. Y. Wang, "Blockchain: The state of the art and future trends," *Acta Automatica Sinica*, vol. 42, no. 4, pp. 481–494, 2016.

[22] J. Chiefari, Y. K. Chong and F. Ercole, "Living free-radical polymerization by reversible addition fragmentation chain transfer: The RAFT process," *Macromolecules*, vol. 31, no. 16, pp. 5559–5562, 1998.

[23] R. F. Yu, "Research on information security technology of Internet of things based on blockchain," M.S. Theses, University of Electronic Science and Technology, China, 2019.

[24] A. Bahga and V. K. Madisetti, "Blockchain platform for industrial Internet of Things," *Journal of Software Engineering and Applications*, vol. 9, no. 10, pp. 533–546, 2016.